

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE  
BEFORE THE BOARD OF PATENT APPEALS AND INTERFERENCES

RECEIVED  
CENTRAL FAX CENTER  
DEC 20 2004

In re Application of:  
Broussard

Serial No. 09/870,614

Filed: May 31, 2001

For: DYNAMIC BUFFERING OF GRAPHIC  
IMAGES BY A PLATFORM  
INDEPENDENT APPLICATION  
PROGRAM INTERFACE

Group Art Unit: 2173  
Examiner: Bonshock, D.

Atty. Dkt. No. AUS920010265US1

I hereby certify that this correspondence is being transmitted via  
facsimile or deposited with the U.S. Postal Service with  
sufficient postage as First Class Mail in an envelope addressed  
to: Commissioner for Patents, Washington, D.C. 20231, on the  
date indicated below:

12/20/2004  
Date

Kevin L. Daffier

APPEAL BRIEF

Box AF  
Commissioner for Patents  
Washington D.C. 20231

Sir/Madam:

Further to the Notice of Appeal faxed October 26, 2004 and received in the U.S. Patent and Trademark Office on the same day, Appellant presents this Appeal Brief. The Notice of Appeal was filed following mailing of a final Office Action on August 13, 2004. Appellant hereby appeals to the Board of Patent Appeals and Interferences from a final rejection of claims 1-22 in the Final Office Action, and respectfully requests that this appeal be considered by the Board.

1. REAL PARTY IN INTEREST

The subject application is owned by International Business Machines Corporation, a corporation having its principal place of business at New Orchard Road, Armonk, New York, 10504, as evidenced by the assignment recorded at Reel 011888, Frame 0549.

## **II. RELATED APPEALS AND INTERFERENCES**

Notices of Appeal have been filed for the following applications, which share a common specification with the application currently on appeal.

09/870,615: Notice of Appeal filed 9/14/04; Appeal Brief filed November 9, 2004.

09/870,621: Notice of Appeal filed 9/24/04; Appeal Brief filed November 23, 2004.

09/870,622: Notice of Appeal filed 8/24/04; Appeal Brief filed October 25, 2004.

Application serial numbers 09/870,615 and 09/870,622 share similar cited art references with the present application; however, dissimilar art is cited in the present application and application serial number 09/870,621. No other appeals or interferences are known which would directly affect or be directly affected by or have a bearing on the Board's decision in this appeal.

## **III. STATUS OF CLAIMS**

Claims 1-19 were originally filed in the present application. Claims 1, 2, 5, 8, 10, 11, 15, 16, and 19 were amended in a response filed April 29, 2004 to an Office Action mailed January 29, 2004. Claims 20-22 were added in the same response. Claims 1, 2, 5 and 6 stand finally rejected under 35 U.S.C. §102, and claims 3, 4 and 7-22 stand finally rejected under 35 U.S.C. §103, which are the subject of this appeal. A copy of claims 1-22, as on appeal (incorporating entered amendments), is included in the Appendix hereto.

## **IV. STATUS OF AMENDMENTS**

No amendments to the claims were filed subsequent to their final rejection. The Appendix hereto therefore reflects the current state of the claims.

## **V. SUMMARY OF THE INVENTION**

Appellant's claimed invention relates to a display system (10, Fig. 1), a computer-readable storage device (18, Fig. 1) and a method (Fig. 15) for displaying a graphical representation (26, Fig. 2) of an object (24). In accordance with aspects of the present invention, however, the graphical representation of the object may, in some cases, be temporarily stored within a display buffer (e.g.,

memory 18 or processor 12 of Fig. 1) before the graphical representation is forwarded to a display device (16, Fig. 1). As described in more detail below, the Appellant's claimed invention more specifically relates to a system and method for enabling/disabling buffering of the graphical representation. (Specification – page 34, line 12 to page 35, line 25, and Abstract).

In some embodiments, the presently claimed display system may include a display (16, Fig. 1), a display buffer (included within, e.g., memory 18 or processor 12) coupled to the display, and a processor (12). The processor may be adapted to execute an application program (APP 28, Fig. 2) which, when executed, produces images (26) upon the display. These so-called "images" may otherwise be referred to herein as graphical representations of an object (24). The object may be included within, e.g., a graphical user interface (GUI, Fig. 1) associated with the application program. During a first mode, the images may be forwarded in sequence to the display. During a second mode, however, at least two of the images may be compiled as a combination image (e.g., combined images 228, 230, 232 and 234, Fig. 14) and presented to the display buffer before the combination image is forwarded to the display.

In some cases, the application program may function to disable or enable buffering of the images by configuring the processor to execute in either the first or the second mode (see, Fig. 15). For example, the processor may be configured for executing in the first mode when the display is directly coupled to the processor, or in the second mode, when the display is remotely coupled to the processor.

In some embodiments, the presently claimed computer-readable storage device may include an operating system (OS 40, Fig. 2), an application program (APP 28) running on code compatible with the operating system, and a software component (API 22) invoked by the application program to display object code (24) which, when executed, produces a sequence of images (26) upon a display screen (16, Fig. 1). In some cases, the operating system may include a Windows, Unix or OS/2 computer operating system. The software component can be configured during runtime of the application program to enable or disable buffering of the sequence of images as a combination image before the combination image is sent to the display screen (see, Fig. 15).

In other embodiments, the presently claimed computer-readable storage device may include a windows-based operating system (OS 40, Fig. 2), an application program (APP 28) running under the operating system, an object (24) created at runtime by the application program, and an interface (API 22) independent of the operating system. The interface may be generally adapted for: (i) creating a graphical

representation (26) of the object, and (ii) enabling or disabling buffering of the graphical representation of the object to a memory storage area (e.g., a display buffer) prior to displaying the graphical representation, as directed by the application program (see, Fig. 15).

In some cases, a software component (250, Fig. 15) associated with the interface may be adapted to: (i) determine whether the application program is operating in a remote or direct mode, and (ii) create a peer component (252) to enable or disable buffering of the graphical representation of the object based on the determination made by the software component. For example, the software component may determine that the application program is operating in a remote mode, and as a result, may create a peer component that functions to enable the buffering feature. On the other hand, the software component may create a peer component to disable buffering, if the software component determines that the application program is operating in a direct mode.

In some embodiments, the presently claimed method may include the steps of providing an application program running under an operating system, creating a graphical representation of the object using an interface independent of the operating system, and enabling or disabling buffering of said graphical representation during runtime, as directed by the application program, and before the graphical representation is forwarded to a display device. As shown in Fig. 15, image buffering may be enabled if the display device is remotely coupled to the system running the application program, and disabled otherwise.

## VI. ISSUES

1. Whether claims 1, 2, 5 and 6 are unpatentable under 35 U.S.C. §102(b) as anticipated by U.S. Patent No. 4,951,229 to DiNicola et al. (hereinafter "DiNicola").
2. Whether claims 3, 4 and 7-10 are unpatentable under 35 U.S.C. §103(a) over DiNicola in view of a web publication written by Amy Fowler entitled "*Mixing Heavy and Light Components*" (hereinafter "Fowler").
3. Whether claims 11-13, 18 and 19 are unpatentable under 35 U.S.C. §103(a) over DiNicola in view of a web publication written by Sun Microsystems entitled "*Introducing Swing*" (hereinafter "Sun").
4. Whether claims 14-17 are unpatentable under 35 U.S.C. §103(a) over DiNicola, Fowler and Sun.
5. Whether claims 20-22 are unpatentable under 35 U.S.C. §103(a) over DiNicola.

## **VII. GROUPING OF CLAIMS**

Claims 1-11, 20 and 21 (Group I) stand or fall together.

Claims 12-19 and 22 (Group II) stand or fall together.

The reasons why the two groups of claims are believed to be separately patentable are explained below in the appropriate parts of the Argument.

## **VIII. ARGUMENT**

Swing was developed as part of the Java Foundation Classes in an effort to overcome the platform-dependency of AWT-based (i.e., Java) application programs. An application program interface (API) written using Swing contains no native code, and therefore, can be run on substantially any OS without changing the look and feel of the application. *See, e.g., Specification, page 9, lines 14-30, and page 19, line 25 to page 20, line 4.* Unfortunately, conventional Swing-based APIs lack many desirable features originally included within the AWT implementation, one of which concerns the manner in which images are buffered before they are sent to a display. *See, e.g., Specification, page 10, lines 17-19.*

In other words, conventional Swing-based APIs default to buffering its output. For example, instead of directly displaying graphics content, the graphics content is first drawn to an image buffer, and the entire buffer is then displayed *en masse*. This is shown in Fig. 14, where Frame 220 contains Panel 222, which in turn contains two Buttons 224 and 226. When the Frame and its contents are to be displayed, they are first drawn to an image buffer where the images of the objects (e.g., the Frame, Panel and Buttons) are combined into a composite image. When combined, bitmaps of overlying components (e.g., Buttons 224 and 226) may replace portions of the bitmap of an underlying component (e.g., certain portions of panel 222). *See, e.g., Specification, page 34, lines 12-26 and Fig. 14.*

Depending on the circumstances, the default buffering of Swing images may lead to sub-optimal performance. As stated in the Specification, for example, "if several Swing objects are present within multiple levels of containment in a layout, the overhead associated with buffering may defeat the hardware acceleration capability of video graphics hardware within the computer. This would then result in a slower-than-necessary display of the objects. On the other hand, buffering may be advantageous in a system involving remote displays. Rather than sending the graphic representation for each object in the layout to a remote terminal in the sequence in which the objects are drawn, it may be faster in such cases

to draw the entire layout to a local buffer, and then send the buffered contents in a single transmission, thereby minimizing network traffic and bandwidth." Specification, page 34, line 28 to page 35, line 7.

For these reasons, the presently claimed case recognized a need to provide an improved system and method that optimizes performance by allowing Swing output buffering to be dynamically enabled or disabled before the output is sent to a display. See, e.g., Specification, page 35, lines 9-10.

The invention as recited in claims 1-22 addresses the above-mentioned need by providing a display system, software components and method for dynamically enabling/disabling image buffering before the images are sent to the display. If buffering is disabled, the images may be forwarded in sequence to the display. If buffering is enabled, however, at least two of the images may be compiled as a combination image, which is presented to a display buffer before it is forwarded to the display. In some cases, the determination of whether to enable or disable buffering may depend on whether a display device is locally or remotely coupled to the display system.

In one particular embodiment, one or more software components (e.g., AWT Swing Peer 252 and JComponent 254) may be added to the AWT Swing toolkit (250) to provide the capability for enabling buffering in situations where it will enhance graphics throughput. For example, the one or more software components may determine whether images are to be displayed on a local display device, or on a display device that is physically remote from the system (e.g., accessible over a network). Based on such determination, buffering may be automatically enabled or disabled by the one or more software components without intervention by the application programmer or modification to the application program code. See, e.g., Specification, page 35, lines 10-25 and Fig. 15.

As described in more detail below, none of the cited art, either separately or in combination, provides motivation to teach or suggest the presently claimed display system, software component or method for dynamically enabling/disabling image buffering before the images are sent to a display. Therefore, the teachings of the cited art cannot be used to render the limitations of the presently claimed case unpatentable.

**ISSUE 1 ARGUMENTS****Patentability of Group I Claims 1, 2, 5 and 6:**

1. **DiNicola fails to disclose a display system including a processor adapted to produce images upon a display, wherein during a second mode, at least two of said images are compiled as a combination image and presented to a buffer before being forwarded to the display.**

Independent claim 1 states in part:

A display system, comprising: a display; a display buffer ... and a processor adapted to execute an application program which, when executed, produces images upon the display ... wherein during a second mode the images are compiled as a combination image of at least one of said images drawn over at least another of said images and presented to the buffer before being forwarded to the display.

Independent claim 5 recites a similar limitation by disclosing a software component that can be configured to enable or disable buffering of a sequence of images as a combination image before sending the combination image to the display.

DiNicola discloses an apparatus and method for managing multiple images in a graphic display system (DiNicola, Title). On page 2 of the Office Action, the Examiner suggests, "DiNicola teaches, in column 2, line 59 through column 3, line 16, especially column 3, lines 10-13, a system that can be configured to... combine two or more of the images and send them as a composite display image." The Examiner further suggests that DiNicola provides teaching for "presenting the image to the buffer before forwarding... in column 3, lines 62-68..." (Office Action, page 3). Though DiNicola may "combine two or more of the images to form a composite display image" (DiNicola, column 3, lines 10-13), as pointed out by the Examiner, DiNicola does not disclose that the composite display image may be presented to a display buffer before it is forwarded to the display, as taught in present claims 1 and 5.

In the passage cited by the Examiner (e.g., DiNicola, column 3, lines 62-68), DiNicola describes how output messages from processor 10 may be formatted and stored in memory buffers 24, 26, 28 and 30, as shown in Fig. 1 of DiNicola. The formatted output messages of DiNicola, however, cannot be considered equivalent to the presently claimed "combination image", since DiNicola's image mixing

process is performed downstream of memory buffers 24, 26, 28 and 30. For example, DiNicola states, "[i]mage mixer 32 reads the encoded graphics data from the memory buffers 24, 26, 28 and 30 and performs the necessary decoding and image mixing ... The merged images are passed through the color translation table 34 which generates the appropriate control signals to be passed on data line 40 to display monitor 50." (DiNicola, column 4, lines 14-30; and see, Fig. 1 of DiNicola). DiNicola does not disclose that the result of the image mixing process (i.e., the composite display image) may be stored within memory buffers 24, 26, 28 and 30, or within any other memory buffer, before the composite display image is forwarded to the display. In other words, DiNicola does not disclose the presently claimed process of enabling/disabling the buffering of a combination image before that image is sent to the display. As a consequence, the teachings of DiNicola do not anticipate all limitations of present claims 1 and 5.

The above-mentioned argument was submitted in a Response mailed April 29, 2004 to the Office Action mailed January 29, 2004. After considering the above argument, the Examiner suggests that DiNicola provides teaching for the limitations of present claim 1 by disclosing "in column 2, line 66 through column 3, line 16 and in column 5, lines 18-27, an intermediate buffer that he doesn't require as a means to make the system more efficient." (Final Office Action, page 17). The Examiner continues to suggest that the "intermediate buffer [is] located after the mixing step," even though it is not required (Final Office Action, page 17). With regard to present claim 5, the Examiner suggests that DiNicola provides teaching for "optional buffering of the sequence of images as a combination image before sending the combination image to the display" by disclosing (e.g., in column 3, lines 62-68 and column 5, lines 18-32) "an intermediate buffer that is not required, as a matter of efficiency, but mentioned in the reference." (Final Office Action, page 4). As will be described in more detail below, the graphical display system of DiNicola does not include an intermediate buffer, nor is there sufficient motivation within DiNicola that would enable one skilled in the art to modify the display system of DiNicola to include an intermediate buffer (e.g., between image mixer 32 and display monitor 50).

As evidenced in Fig. 1, DiNicola does not include an intermediate frame buffer between image mixer 32 and display monitor 50. Furthermore, DiNicola does not teach, suggest or even provide sufficient motivation for the intermediate frame buffer being an "optional" feature of the invention, as suggested by the Examiner. Instead, DiNicola describes the intentional absence of an intermediate frame buffer as an advantage over prior art display systems. Therefore, DiNicola appears to teach away from presenting a combination image to a display buffer before the combination image is forwarded to the



display device. As noted in section 2143.02 of the MPEP, a prior art reference must be considered in its entirety, i.e., as a whole, including portions that would lead away from the claimed invention. *W.L. Gore & Associates, Inc. v. Garlock, Inc.*, 721 F.2d 1540, 220 USPQ 303 (Fed. Cir. 1983). Appellants assert that a prior art reference cannot be used to anticipate a claimed limitation if that reference specifically teaches away from the limitation.

DiNicola specifically teaches away from the use of an intermediate frame buffer by stating, "[t]his invention ... provides an inventive capability to combine images from separate lateral bit encoded bit planes. Image mixing is performed by image mixer 32 which reads data from the bit planes, combines the image data and transmits it directly to the display monitor. There is no requirement for an intermediate frame buffer to store the results of the image mixing operation. This is a significant advantage because a change to one of the bit planes does not require the complete regeneration of an intermediate frame buffer." (DiNicola, column 5, lines 18-27, emphasis added). By specifically stating that there is no requirement for an intermediate frame buffer, DiNicola provides absolutely no teaching, suggestion or even motivation for including the intermediate frame buffer as an "optional" feature.

Therefore, Appellants respectfully traverse the Examiner's contention that the mere mention of an intermediate frame buffer within the cited art reference would somehow motivate one skilled in the art to modify the graphical display system of DiNicola to include such a buffer. One skilled in the art would not be motivated to include an intermediate frame buffer within the graphical display system of DiNicola, since DiNicola specifically teaches away from doing so. Furthermore, Appellants assert that even if the graphical display system of DiNicola were somehow modified to include the so-called "optional" display buffer (without sufficient motivation to do so), DiNicola would still lack teaching or suggestion for enabling the buffering capability during a first mode and disabling the buffering capability during a second mode. Therefore, Appellants assert that not only does DiNicola fail to anticipate all limitations of present claims 1 and 5, DiNicola cannot be modified to do so.

**2. The Examiner has failed to support a ground of anticipation of claims 1 and 5 by DiNicola.**

The standard for "anticipation" is one of fairly strict identity. A claim is anticipated only if each and every element as set forth in the claim is found, either expressly or inherently described, in a single prior art reference. *Verdegal Bros. v. Union Oil Co. Of California*, 2 USPQ2d 1051, 1053 (Fed. Cir. 1987); MPEP 2131. As pointed out above, DiNicola does not disclose a "display system, comprising ...

a processor adapted to execute an application program which, when executed, produces images upon the display ... wherein during a second mode the images are compiled as a combination image of at least one of said images drawn over at least another of said images and presented to the buffer before being forwarded to the display," as recited in present claim 1. In addition, DiNicola does not disclose a "computer-readable memory, comprising ... a software component invoked by the application program to display object code which, when executed, produces a sequence of images upon a display screen, wherein the software component can be configured during runtime of the application program to enable or disable buffering of the sequence of images as a combination image before sending the combination image to the display," as recited in present claim 5. DiNicola, therefore, cannot teach each and every element set forth in claims 1 and 5. As a consequence, claims 1 and 5 are not anticipated by DiNicola.

### **Conclusion**

As explained in Arguments 1-2 above, several limitations of independent claims 1 and 5 are not taught or suggested by DiNicola. Therefore, claims 1 and 5, as well as claims dependent therefrom, are not anticipated by DiNicola. Therefore, the rejection of Group I claims 1, 2, 5 and 6 under 35 U.S.C. § 102 is asserted to be erroneous.

### **ISSUE 2 ARGUMENTS**

#### **Patentability of Group I Claims 3, 4 and 7-10:**

Because claims 3, 4 and 7-10 of Group I are dependent from claims 1 and 5, the arguments presented above for patentability of claims 1 and 5 apply equally to claims 3, 4 and 7-10, and are herein incorporated by reference. In addition to the 35 U.S.C. § 102 arguments presented above with respect to claims 1 and 5, arguments are provided below to establish patentability of the current claims under 35 U.S.C. § 103(a).

- 1. DiNicola and Fowler each fail to disclose a display system including a processor adapted to produce images upon a display, wherein during a second mode, at least two of said images are compiled as a combination image and presented to a buffer before being forwarded to the display.**

The above-mentioned limitation relates to independent claim 1; however, independent claim 5 recites a similar limitation by disclosing a software component that can be configured to enable or disable the buffering of a sequence of images as a combination image before the combination image is sent to the display. As described in the section 102 arguments presented above, DiNicola simply fails to provide teaching, suggestion, or even motivation for the presently claimed display system (claim 1) and software component (claim 5).

Appellants realize that Fowler is not relied upon for teaching the limitations of present claims 1 and 5, but instead, is relied upon for mentioning certain AWT and Swing components (e.g., frames, panels and buttons). However, since currently rejected claims 3, 4 and 7-10 are dependent on claims 1 and 5 and are, therefore, patentable for at least the same reasons as those claims, Appellants wish to address the lack of teaching within Fowler for the limitations of independent claims 1 and 5.

In a publication entitled *Mixing Heavy and Light Components*, Fowler provides a brief overview of certain problems that may be encountered when mixing heavy (e.g., AWT-based) and light (e.g., Swing-based) components, and presents certain guidelines that should be followed to circumvent such problems. However, Fowler fails to mention that images generated by the Swing components (e.g., images of frames, panels and buttons) could be buffered. Unlike the presently claim case, Fowler fails to mention the manner in which Swing-generated images are automatically buffered before the images can be sent to a display device, and the potential problems associated with default buffering. Since Fowler fails to mention a display buffer, buffering of images, or any means or desirability for enabling/disabling a display buffer, Fowler cannot be relied upon to provide teaching or suggestion for the limitations recited in present claims 1 and 5.

**2. There is no motivation to modify or combine the teachings of DiNicola and Fowler to provide the presently claimed display system and software components**

As described above in the arguments presented for the patentability of Group I claims 1 and 5, DiNicola cannot be modified to teach or suggest the presently claimed display system and software components, which allow a sequence of images to be compiled as a combination image and presented to a display buffer before the combination image is forwarded to a display device. As will be described in more detail below, Fowler cannot be modified or combined with DiNicola to overcome the deficiencies therein.

Obviousness can only be established by combining or modifying the teachings of the prior art to produce the claimed invention where there is some teaching, suggestion, or motivation to do so. *In re Fine*, 837 F.2d 1071, 5 USPQ2d 1596 (Fed.Cir. 1988); *In re Jones*, 958 F.2d 347, 21 USPQ2d 1941 (Fed. Cir. 1992); MPEP 2143.01.

As noted above, Fowler provides absolutely no teaching or suggestion for a display buffer, buffering of images, or any means for enabling/disabling a display buffer. Though Fowler mentions a few of the problems encountered when mixing Swing and AWT components, Fowler does not mention the potential problems associated with the default buffering of Swing-generated images. For example, Fowler does not suggest that buffering may unnecessarily slow down the display of images, when those images are to be displayed on a local display device. Therefore, Fowler provides no motivation for enabling and disabling the buffering of images, and as a consequence, cannot be modified to do so.

Since DiNicola and Fowler each fail to teach, suggest or provide motivation for enabling and disabling the buffering of images, DiNicola and Fowler cannot be combined to do so. Even if the proposed combination were made (without sufficient motivation to do so), Appellants assert that the combined teachings of DiNicola and Fowler would still fail to disclose all limitations of present claims 1 and 5.

**3. The Examiner has failed to adequately support and/or establish a *prima facie* ground of obviousness.**

To establish a *prima facie* case of obviousness, three basic criteria must be met. First, there must be some suggestion or motivation, either in the references themselves or in the knowledge generally available to one of ordinary skill in the art, to modify the reference or to combine reference teachings. Second, there must be a reasonable expectation of success. Finally, the prior art reference (or references when combined) must teach or suggest all claim limitations. MPEP § 2143. None of these three criteria have been met by the Examiner in the present case. First of all, no suggestion or motivation to modify the cited references can be found within the cited references to teach or suggest the aforementioned limitations of claims 1 and 5, as explained above in Argument 2. The criterion of a reasonable expectation of success cannot be met if no teaching, suggestion or motivation exists, because there is then nothing at which to be successful. Finally, none of the cited art, either alone or in combination, teaches all of the limitations of claims 1 and 5, as explained above in Arguments 1 and 2. The third

criterion recited above has therefore also not been met, and a *prima facie* case of obviousness has not been established.

### **Conclusion**

As explained in Arguments 1-3 above, at least some limitations of claims 1 and 5 and, therefore, at least some limitations of claims 3, 4, and 7-10, are not taught or suggested by the cited art. Furthermore, there is no teaching, suggestion or motivation to modify the cited art to teach the limitations of these claims. For at least the reasons set forth above, claims 3, 4, and 7-10 are patentably distinct over the cited art. Therefore, the rejection of Group I claims 3, 4, and 7-10 under 35 U.S.C. § 103(a) is asserted to be erroneous.

### **ISSUE 3 ARGUMENTS**

#### **Patentability of Group I Claim 11:**

Because claim 11 of Group I is dependent from claim 5, the arguments presented above for patentability of claim 5 apply equally to claim 11, and are herein incorporated by reference. In addition to the 35 U.S.C. § 102 arguments presented above with respect to claim 5, arguments are provided below to establish patentability of the current claims under 35 U.S.C. § 103(a).

- 1. DiNicola and Sun each fail to disclose a software component, which is configured to enable or disable buffering of a sequence of images as a combination image before the combination image is sent to a display.**

The above-mentioned limitation relates to independent claim 5. As described in the section 102 arguments presented above, DiNicola simply fails to provide teaching, suggestion, or motivation for the software component recited in present claim 5.

Appellants realize that Sun is not relied upon for teaching the limitations of present claim 5, but instead, is relied upon for describing how Swing-based application programs can be run on different operating systems (such as Windows and Unix). However, since currently rejected claim 11 is dependent on claim 5 and is, therefore, patentable for at least the same reasons as claim 5, Appellants wish to address the lack of teaching within Sun for the limitations of independent claim 5.

The article entitled *Introducing Swing* does just as the title implies – it briefly outlines the Swing architecture and the basic differences between Swing and AWT. However, *Introducing Swing* (otherwise referred to as SUN) fails to provide teaching or suggestion for the software component recited in present claim 5, which is configured to enable or disable buffering of a sequence of images as a combination image before the combination image is sent to a display. In addition, Sun fails to mention the use of display buffers, buffering of images, the manner in which Swing-generated images are automatically buffered by default, or the potential problems associated with default buffering. As a consequence, Sun cannot be relied upon to provide teaching or suggestion for a software component specifically configured to enable or disable buffering of a sequence of images as a combination image before the combination image is sent to a display.

**2. There is no motivation to modify or combine the teachings of DiNicola and Sun to provide the presently claimed software component.**

As described above in the arguments presented for the patentability of Group I claims 1 and 5, DiNicola cannot be modified to teach or suggest the presently claimed software component, which allows a sequence of images to be compiled as a combination image and presented to a display buffer before the combination image is forwarded to a display device. As will be described in more detail below, Sun cannot be modified or combined with DiNicola to overcome the deficiencies therein.

Obviousness can only be established by combining or modifying the teachings of the prior art to produce the claimed invention where there is some teaching, suggestion, or motivation to do so. *In re Fine*, 837 F.2d 1071, 5 USPQ2d 1596 (Fed.Cir. 1988); *In re Jones*, 958 F.2d 347, 21 USPQ2d 1941 (Fed. Cir. 1992); MPEP 2143.01.

As noted above, Sun provides absolutely no teaching or suggestion for a display buffer, buffering of images, or any means (e.g., a software component) for enabling/disabling a display buffer. In addition, and contrary to the presently claimed case, Sun does not disclose the manner in which Swing-generated images are automatically buffered by default, or the potential problems associated with default buffering. Therefore, Sun provides no motivation that would enable one skilled in the art to modify the teachings of Sun to include a software component, which is capable of enabling/disabling the buffering of a sequence of images as a combination image before the combination image is sent to a display, as recited in present claim 5. As a consequence, Sun cannot be modified to provide teaching or suggestion for the limitations of present claim 5.

However, statements in the Office Action mailed January 29, 2004 and in the final Office Action mailed April 29, 2004 suggest that "DiNicola teaches the system that either transmits images to the display sequentially or as a combination image ... [and] Sun teaches Swing[,], which is stated in the specification of the application to default to buffering it's output, similar to that of DiNicola ..." (Office Action, page 8; Final Office Action, page 9). As such, the Examiner appears to rely on the Applicant's own disclosure to provide motivation for combining the teachings of DiNicola and Sun. This simply cannot be done. As noted in section 2143 of the MPEP, the teaching or suggestion to make the claimed combination and the reasonable expectation of success must both be found in the prior art, not in applicant's disclosure. *In re Vaeck*, 947 F.2d 488, 20 USPQ2d 1438 (Fed. Cir. 1991).

As noted above, DiNicola and Sun each fail to teach, suggest or provide motivation for enabling and disabling the buffering of images. In addition, DiNicola and Sun cannot be modified or combined to provide such teaching, since neither DiNicola nor Sun suggest even the slightest desirability for making the proposed combination or modification. Though the Applicant's disclosure mentions Swing's default buffering capability (which may be considered an inherent aspect of Swing), the potential problems associated therewith and the particular solution provided by the presently claimed case would not be apparent to one skilled in the art given the teachings of Sun and DiNicola. In other words, the Applicant's own disclosure cannot be used as motivation to combine or modify the prior art references when motivation cannot be found in the prior art references themselves.

**3. The Examiner has failed to adequately support and/or establish a *prima facie* ground of obviousness.**

To establish a *prima facie* case of obviousness, three basic criteria must be met. First, there must be some suggestion or motivation, either in the references themselves or in the knowledge generally available to one of ordinary skill in the art, to modify the reference or to combine reference teachings. Second, there must be a reasonable expectation of success. Finally, the prior art reference (or references when combined) must teach or suggest all claim limitations. MPEP § 2143. None of these three criteria have been met by the Examiner in the present case. First of all, no suggestion or motivation to modify the cited references can be found within the cited references to teach or suggest the aforementioned limitations of claim 5, as explained above in Argument 2. The criterion of a reasonable expectation of success cannot be met if no teaching, suggestion or motivation exists, because there is then nothing at which to be successful. Finally, none of the cited art, either alone or in combination, teaches all of the

limitations of claim 5, as explained above in Argument 1. The third criterion recited above has therefore also not been met, and a *prima facie* case of obviousness has not been established.

### **Conclusion**

As explained in Arguments 1-3 above, at least some limitations of claim 5 and, therefore, at least some limitations of claim 11, are not taught or suggested by the cited art. Furthermore, there is no teaching, suggestion or motivation to modify the cited art to teach the limitations of these claims. For at least the reasons set forth above, claim 11 is patentably distinct over the cited art. Therefore, the rejection of Group I claim 11 under 35 U.S.C. § 103(a) is asserted to be erroneous.

### **Patentability of Group II Claims 12, 13, 18, and 19:**

1. DiNicola and Sun each fail to disclose a computer-readable storage device (claim 19) and method (claim 12) for displaying an object by enabling or disabling buffering of a graphical representation of the object to a memory storage area prior to displaying the graphical representation.

Independent claim 12 recites, in part:

A method for displaying an object, comprising ...creating a graphical representation of the object using an interface independent of the operating system; and enabling or disabling buffering of said graphical representation during runtime as directed by the application program.

Independent claim 19 recites, in part:

A computer-readable storage device, comprising ... an interface independent of the operating system, and adapted for: creating a graphical representation of the object; and enabling or disabling buffering of the graphical representation of the object to a memory storage area prior to displaying the graphical representation, as directed by the application program.

As described in the section 102 and 103 arguments presented above, DiNicola simply fails to provide teaching, suggestion, or motivation for the limitations of claims 1 and 5, which relate to enabling/disabling the buffering of a sequence of images as a combination image before the combination image is forwarded to a display.



According to a more general embodiment of the invention, present claims 12 and 19 each recite limitations on enabling/disabling the buffering of a "graphical representation of an object." This is also not taught or suggested by DiNicola. For example, the only instance in which DiNicola actually utilizes image buffering occurs when "[o]utput messages from the processor 10 are formatted and stored in memory buffers, or bit planes 24, 26, 28, and 30." (DiNicola, column 3, lines 62-66). However, DiNicola does not teach or suggest that memory buffers 24, 26, 28 and 30 could be disabled in certain circumstances. Since no other memory buffers exist within the graphical display system of DiNicola (*see* above arguments as to why the "intermediate frame buffer" of DiNicola does not and cannot be used to provide "optional buffering," as suggested by the Examiner), DiNicola cannot be relied upon to provide teaching or suggestion for the presently claimed limitations of enabling/disabling the buffering of a graphical representation of an object.

Sun is not relied upon to provide teaching or suggestion for enabling/disabling the buffering of a graphical representation of an object. However, Appellants assert that no teaching or suggestion can be found within Sun for such a limitation.

**2. There is no motivation to modify or combine the teachings of DiNicola and Sun to provide the presently claimed computer-readable storage device or method.**

DiNicola fails to provide motivation for the presently claimed computer-readable storage device and method by failing to even suggest that memory buffers 24, 26, 28 and 30 could be disabled in certain circumstances. In other words, there is simply no motivation within DiNicola that would enable one skilled in the art to make such modification. Furthermore, if the teachings of DiNicola were somehow modified (without sufficient motivation to do so) to provide the capability for enabling/disabling memory buffers 24, 26, 28 and 30, Appellants contend that the modified invention of DiNicola would be rendered unsatisfactory for its intended purpose. As noted in section 2143.01 of the MPEP, if the proposed modification would render the prior art invention being modified unsatisfactory for its intended purpose, then there is no suggestion or motivation to make the proposed modification. *In re Gordon*, 733 F.2d 900, 221 USPQ 1125 (Fed. Cir. 1984).

For example, the intended purpose of DiNicola is to support both bit encoded and multiplane lateral bit encoding techniques. (*See*, DiNicola, column 2, lines 49-58). DiNicola states that, in bit plane encoding, "[t]he information stored in any one plane represents only a subset of the information required to create the picture element on the display monitor. None of the planes represent the full structure of the

image; it is only the combination of the several planes that allows the final image to be made apparent. Data is read from each bit plane simultaneously from the same relative bit location. For example, bits  $c_0$ ,  $c_1$  and  $c_2$  would be read from the bit planes to form a single picture element for display." (DiNicola, column 4, lines 39-48). Therefore, in order to achieve bit plane encoding, DiNicola states that data must be stored in the bit planes (i.e., memory buffers 24, 26, 28 and 30) before individual bits can be "mixed" to form a single picture element for display. Since bit plane encoding would not be possible if memory buffers 24, 26, 28 and 30 were disabled, DiNicola cannot be modified with the capability for enabling and disabling the memory buffers, since such modification would not allow the graphical display system of DiNicola to support both bit encoded and multiplane lateral bit encoding techniques.

In addition, Sun cannot be modified or combined with DiNicola to overcome the deficiencies therein. Obviousness can only be established by combining or modifying the teachings of the prior art to produce the claimed invention where there is some teaching, suggestion, or motivation to do so. *In re Fine*, 837 F.2d 1071, 5 USPQ2d 1596 (Fed.Cir. 1988); *In re Jones*, 958 F.2d 347, 21 USPQ2d 1941 (Fed. Cir. 1992); MPEP 2143.01.

As noted above, Sun provides absolutely no teaching or suggestion for a display buffer, buffering of images, or any means for enabling/disabling image buffering. Though Sun provides a brief overview of Swing, Sun does not disclose the manner in which Swing-generated images are automatically buffered by default, or the potential problems associated with default buffering. Therefore, Sun provides no motivation that would enable one skilled in the art to modify the teachings of Sun to include a computer-readable storage device or method for enabling/disabling the buffering of a graphical representation of an object, as recited in present claims 12 and 19. As a consequence, Sun cannot be modified to provide teaching or suggestion for the aforementioned limitation of present claims 12 and 19.

Since DiNicola and Sun each fail to teach, suggest or provide motivation for enabling and disabling the buffering of images (i.e., graphical representations of objects), DiNicola and Sun cannot be combined to do so. Even if the proposed combination were made (without sufficient motivation to do so), Appellants assert that the combined teachings of DiNicola and Sun would still fail to disclose all limitations of present claims 12 and 19.

**3. The Examiner has failed to adequately support and/or establish a *prima facie* ground of obviousness.**

To establish a *prima facie* case of obviousness, three basic criteria must be met. First, there must be some suggestion or motivation, either in the references themselves or in the knowledge generally available to one of ordinary skill in the art, to modify the reference or to combine reference teachings. Second, there must be a reasonable expectation of success. Finally, the prior art reference (or references when combined) must teach or suggest all claim limitations. MPEP § 2143. None of these three criteria have been met by the Examiner in the present case. First of all, no suggestion or motivation to modify the cited references can be found within the cited references to teach or suggest the aforementioned limitations of claims 12 and 19, as explained above in Argument 2. The criterion of a reasonable expectation of success cannot be met if no teaching, suggestion or motivation exists, because there is then nothing at which to be successful. Finally, none of the cited art, either alone or in combination, teaches all of the limitations of claims 12 and 19, as explained above in Argument 1. The third criterion recited above has therefore also not been met, and a *prima facie* case of obviousness has not been established.

**Conclusion**

As explained in Arguments 1-3 above, at least some limitations of claims 12 and 19 and, therefore, at least some limitations of claims 13 and 18, are not taught or suggested by the cited art. Furthermore, there is no teaching, suggestion or motivation to modify the cited art to teach the limitations of these claims. For at least the reasons set forth above, claims 12, 13, 18, and 19 are patentably distinct over the cited art. Therefore, the rejection of Group II claims 12, 13, 18, and 19 under 35 U.S.C. § 103(a) is asserted to be erroneous.

**ISSUE 4 ARGUMENTS**

**Patentability of Group II Claims 14-17:**

Because claims 14-17 of Group II are dependent from claim 12, the arguments presented above for patentability of claim 12 apply equally to claims 14-17, and are herein incorporated by reference. In addition to the 35 U.S.C. § 103 arguments presented above with respect to claim 12, arguments are provided below to establish patentability of the current claims under 35 U.S.C. § 103(a).

1. **DiNicola, Fowler and Sun each fail to disclose a method for displaying an object by enabling or disabling buffering of a graphical representation of the object.**

As noted in the arguments present above for the patentability of Group II claims 12 and 19, DiNicola and Sun each fail to teach, suggest or provide motivation for enabling/disabling the buffering of images (i.e., a graphical representation of an object), and more specifically, cannot be combined or modified to do so.

As noted in the arguments present above for the patentability of Group I claims 3, 4 and 7-10, Fowler fails to teach, suggest or provide motivation for enabling and disabling the buffering of images, regardless of whether those images form a "combination image," as recited in claims 1 and 5, or merely a "graphical representation of an object", as recited in claims 12 and 19. Fowler simply fails to teach or suggest a display buffer, buffering of images, or any means for enabling/disabling the buffering of images, and therefore, cannot disclose a method for displaying an object, where the method includes enabling or disabling the buffering of a graphical representation of an object.

2. **The teachings of DiNicola, Fowler and Sun cannot be combined or modified to provide the presently claimed method.**

The only reference that even mentions image buffering is DiNicola. However, and as repeatedly stated above, DiNicola does not teach the capability for enabling/disabling the buffering of images, nor can the teachings of DiNicola be modified to do so. Since Fowler and Sun fail to even mention buffering, Fowler and Sun cannot be modified or combined with DiNicola to overcome the deficiencies therein.

3. **The Examiner has failed to adequately support and/or establish a *prima facie* ground of obviousness.**

To establish a *prima facie* case of obviousness, three basic criteria must be met. First, there must be some suggestion or motivation, either in the references themselves or in the knowledge generally available to one of ordinary skill in the art, to modify the reference or to combine reference teachings. Second, there must be a reasonable expectation of success. Finally, the prior art reference (or references when combined) must teach or suggest all claim limitations. MPEP § 2143. None of these three criteria have been met by the Examiner in the present case. First of all, no suggestion or motivation to modify the cited references can be found within the cited references to teach or suggest the aforementioned

limitations of claim 12, as explained above in Argument 2. The criterion of a reasonable expectation of success cannot be met if no teaching, suggestion or motivation exists, because there is then nothing at which to be successful. Finally, none of the cited art, either alone or in combination, teaches all of the limitations of claim 12, as explained above in Argument 1. The third criterion recited above has therefore also not been met, and a *prima facie* case of obviousness has not been established.

### **Conclusion**

As explained in Arguments 1-3 above, at least some limitations of claim 12 and, therefore, at least some limitations of claims 14-17, are not taught or suggested by the cited art. Furthermore, there is no teaching, suggestion or motivation to modify the cited art to teach the limitations of these claims. For at least the reasons set forth above, claims 14-17 are patentably distinct over the cited art. Therefore, the rejection of Group II claims 14-17 under 35 U.S.C. § 103(a) is asserted to be erroneous.

### **ISSUE 5 ARGUMENTS**

#### **Patentability of Group I Claims 20 and 21 and Group II Claim 22:**

Because claims 20 and 21 of Group I and claim 22 of Group II are dependent from claim 1 of Group I and claim 19 of Group II, respectively, the arguments presented above for patentability of claims 1 and 19 apply equally to claims 20-22, and are herein incorporated by reference.

As noted in the arguments present above for the patentability of Group I claim 1 and Group II claim 19, DiNicola simply fails to teach, suggest or provide motivation for enabling/disabling the buffering of images, regardless of whether those images form a "combination image", as recited in claim 1, or merely a "graphical representation of an object", as recited in claim 19. In addition, the above arguments conclusively show that the teachings of DiNicola cannot be combined or modified to provide such teaching.

**Conclusion**


As explained in the above Argument, at least some limitations of claims 1 and 19 and, therefore, at least some limitations of claims 20-22, are not taught or suggested by the cited art. Furthermore, there is no teaching, suggestion or motivation to modify the cited art to teach the limitations of these claims. For at least the reasons set forth above, claims 20-22 are patentably distinct over the cited art. Therefore, the rejections of Group I claims 20-21 and Group II claim 22 under 35 U.S.C. § 103(a) are asserted to be erroneous.

**IX. CONCLUSION**

For the foregoing reasons, it is submitted that the Examiner's rejection of claims 1-22 was erroneous, and reversal of the Examiner's decision is respectfully requested.

The Commissioner is hereby authorized to charge the required fee(s) to deposit account number 50-3268/5468-07800.

Respectfully submitted,

  
Kevin L. Daffer  
Reg. No. 34,146  
Attorney for Appellant

Daffer McDaniel, L.L.P.  
P.O. Box 684908  
Austin, TX 78768-4908  
Date: December 20, 2004

**X. APPENDIX**

The present claims on appeal are as follows.

1. A display system, comprising:

a display;

a display buffer coupled to the display; and

a processor adapted to execute an application program which, when executed, produces images upon the display, wherein during a first mode the images are forwarded in sequence to the display, and wherein during a second mode the images are compiled as a combination image of at least one of said image drawn over at least another of said images and presented to the buffer before being forwarded to the display.

2. The display system as recited in claim 1, wherein the application program disables or enables buffering of the images by configuring the processor to execute in either the first or second mode.

3. The display system as recited in claim 1, wherein the images comprise frame, panel and button images.

4. The display system as recited in claim 1, wherein the application program comprises a Java program.

5. A computer-readable memory, comprising:

an operating system,

an application program running on code compatible with the operating system; and

a software component invoked by the application program to display object code which, when executed, produces a sequence of images upon a display screen, wherein the software component can be configured during runtime of the application program to enable or disable buffering of the sequence of images as a combination image before sending the combination image to the display.

6. The memory as recited in claim 5, wherein the object code is part of a graphical user interface associated with the application program.
7. The memory as recited in claim 5, wherein the software component comprises an application program interface of code which translates between code within the application program and the operating system.
8. The memory as recited in claim 7, wherein a behavior of the application program interface emulates that of a second application program interface based on windowing protocols of a windows-based version of said operating system.
9. The memory as recited in claim 8, wherein the second application program interface comprises a Java abstract windowing toolkit.
10. The memory as recited in claim 5, wherein the application program is written in Java programming language.
11. The memory as recited in claim 5, wherein the operating system comprises a Windows, Unix or OS/2 computer operating system.
12. A method for displaying an object, comprising:
  - providing an application program running under an operating system;
  - creating a graphical representation of the object using an interface independent of the operating system; and
  - enabling or disabling buffering of said graphical representation during runtime as directed by the application program.
13. The method as recited in claim 12, wherein said creating comprises compiling the object as code that includes part of a graphical user interface associated with the application program.



14. The method as recited in claim 12, wherein said creating comprises implementing a call routine to compile a software component that includes an application program interface between the application program and the operating system.

15. The method as recited in claim 14, wherein a behavior of the application program interface emulates that of a second application program interface based on the operating system.

16. The method as recited in claim 15, wherein the second application program interface comprises a Java abstract windowing toolkit.

17. The method as recited in claim 12, wherein the application program is written in Java programming language.

18. The method as recited in claim 12, wherein the operating system comprises a Windows, Unix or OS/2 computer operating system.

19. A computer-readable storage device, comprising:

a windows-based operating system;

an application program running under the operating system;

an object created at runtime by the application program;

an interface independent of the operating system, and adapted for:

creating a graphical representation of the object; and

enabling or disabling buffering of the graphical representation of the object to a memory storage area prior to displaying the graphical representation, as directed by the application program.

20. The display system as recited in claim 2, wherein the processor executes in the first mode when the display is directly coupled to the processor.

21. The display system as recited in claim 2, wherein the processor executes in the second mode when the display is remotely coupled to the processor.

22. The computer-readable storage device as recited in claim 19, further comprising a software component associated with the interface and adapted to: (i) determine whether the application program is operating in a remote or direct mode, and (ii) create a peer component to enable or disable buffering of the graphical representation of the object based on the determination made by the software component.